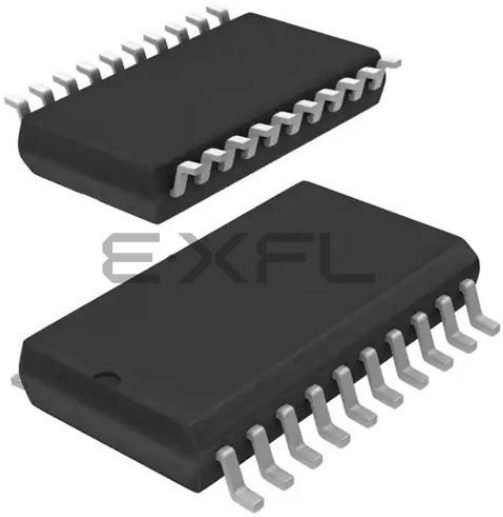


Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Obsolete
Core Processor	8051
Core Size	8-Bit
Speed	20MHz
Connectivity	I²C, UART/USART
Peripherals	Brown-out Detect/Reset, LED, POR, WDT
Number of I/O	18
Program Memory Size	2KB (2K x 8)
Program Memory Type	OTP
EEPROM Size	-
RAM Size	128 x 8
Voltage - Supply (Vcc/Vdd)	2.7V ~ 6V
Data Converters	-
Oscillator Type	Internal
Operating Temperature	0°C ~ 70°C (TA)
Mounting Type	Surface Mount
Package / Case	20-SOIC (0.295", 7.50mm Width)
Supplier Device Package	20-SO
Purchase URL	https://www.e-xfl.com/product-detail/nxp-semiconductors/p87lpc762bd-512

SPECIAL FUNCTION REGISTERS

Name	Description	SFR Address	Bit Functions and Addresses MSB							LSB	Reset Value
			E7	E6	E5	E4	E3	E2	E1	E0	
ACC*	Accumulator	E0h									00h
AUXR1#	Auxiliary Function Register	A2h	KBF	BOD	BOI	LPEP	SRST	0	–	DPS	02h[1]
			F7	F6	F5	F4	F3	F2	F1	F0	
B*	B register	F0h									00h
CMP1#	Comparator 1 control register	ACh	–	–	CE1	CP1	CN1	OE1	CO1	CMF1	00h[1]
CMP2#	Comparator 2 control register	ADh	–	–	CE2	CP2	CN2	OE2	CO2	CMF2	00h[1]
DIVM#	CPU clock divide-by-M control	95h									00h
DPTR:	Data pointer (2 bytes)										
DPH	Data pointer high byte	83h									00h
DPL	Data pointer low byte	82h									00h
			CF	CE	CD	CC	CB	CA	C9	C8	
I2CFG#*	I^2C configuration register	C8h/RD	SLAVEN	MASTRQ	0	TIRUN	–	–	CT1	CT0	00h[1]
		C8h/WR	SLAVEN	MASTRQ	CLRTI	TIRUN	–	–	CT1	CT0	
			DF	DE	DD	DC	DB	DA	D9	D8	
I2CON#*	I^2C control register	D8h/RD	RDAT	ATN	DRDY	ARL	STR	STP	MASTER	–	80h[1]
		D8h/WR	CXA	IDLE	CDR	CARL	CSTR	CSTP	XSTR	XSTP	
I2DAT#	I^2C data register	D9h/RD	RDAT	0	0	0	0	0	0	0	80h
		D9h/WR	XDAT	x	x	x	x	x	x	x	
			AF	AE	AD	AC	AB	AA	A9	A8	
IEN0*	Interrupt enable 0	A8h	EA	EWD	EBO	ES	ET1	EX1	ET0	EX0	00h
			EF	EE	ED	EC	EB	EA	E9	E8	
IEN1#*	Interrupt enable 1	E8h	ETI	–	EC1	–	–	EC2	EKB	EI2	00h[1]
			BF	BE	BD	BC	BB	BA	B9	B8	
IP0*	Interrupt priority 0	B8h	–	PWD	PBO	PS	PT1	PX1	PT0	PX0	00h[1]
IP0H#	Interrupt priority 0 high byte	B7h	–	PWDH	PBOH	PSH	PT1H	PX1H	PT0H	PX0H	00h[1]
			FF	FE	FD	FC	FB	FA	F9	F8	
IP1*	Interrupt priority 1	F8h	PTI	–	PC1	–	–	PC2	PKB	PI2	00h[1]

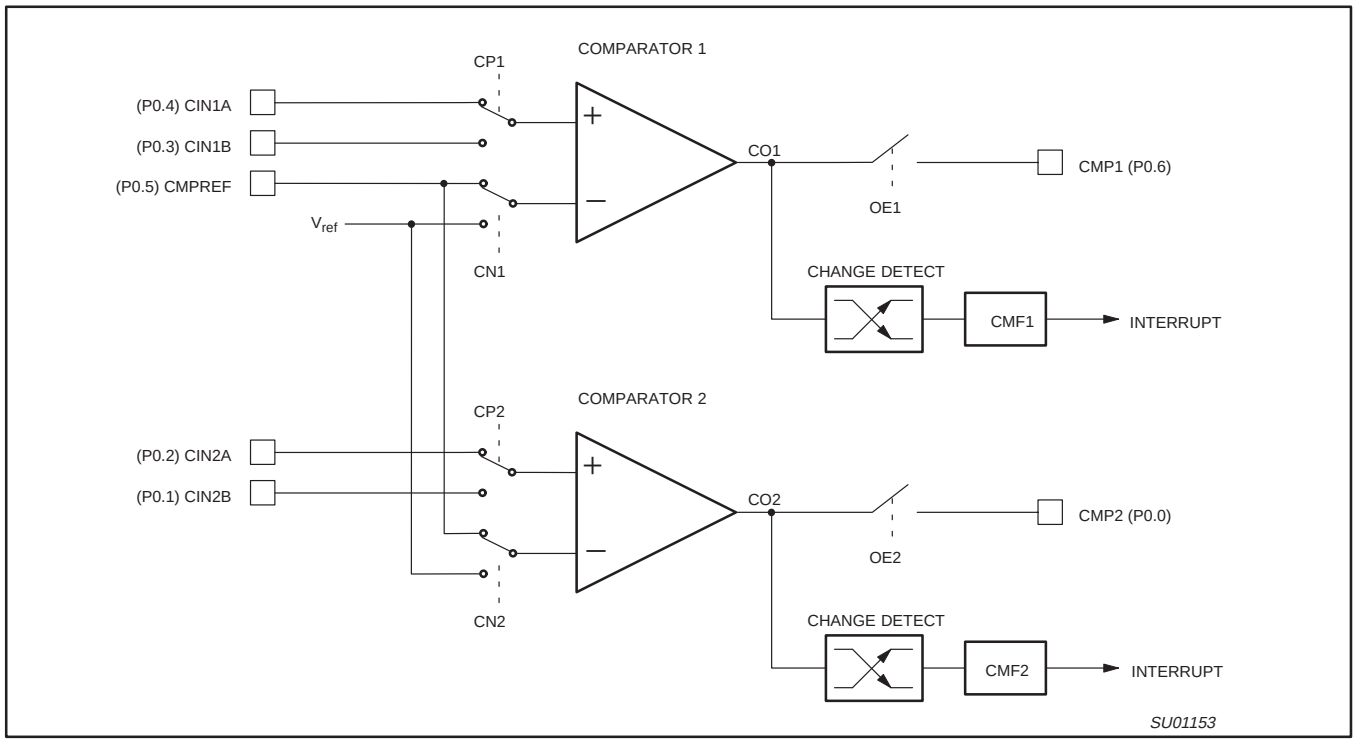


Figure 3. Comparator Input and Output Connections

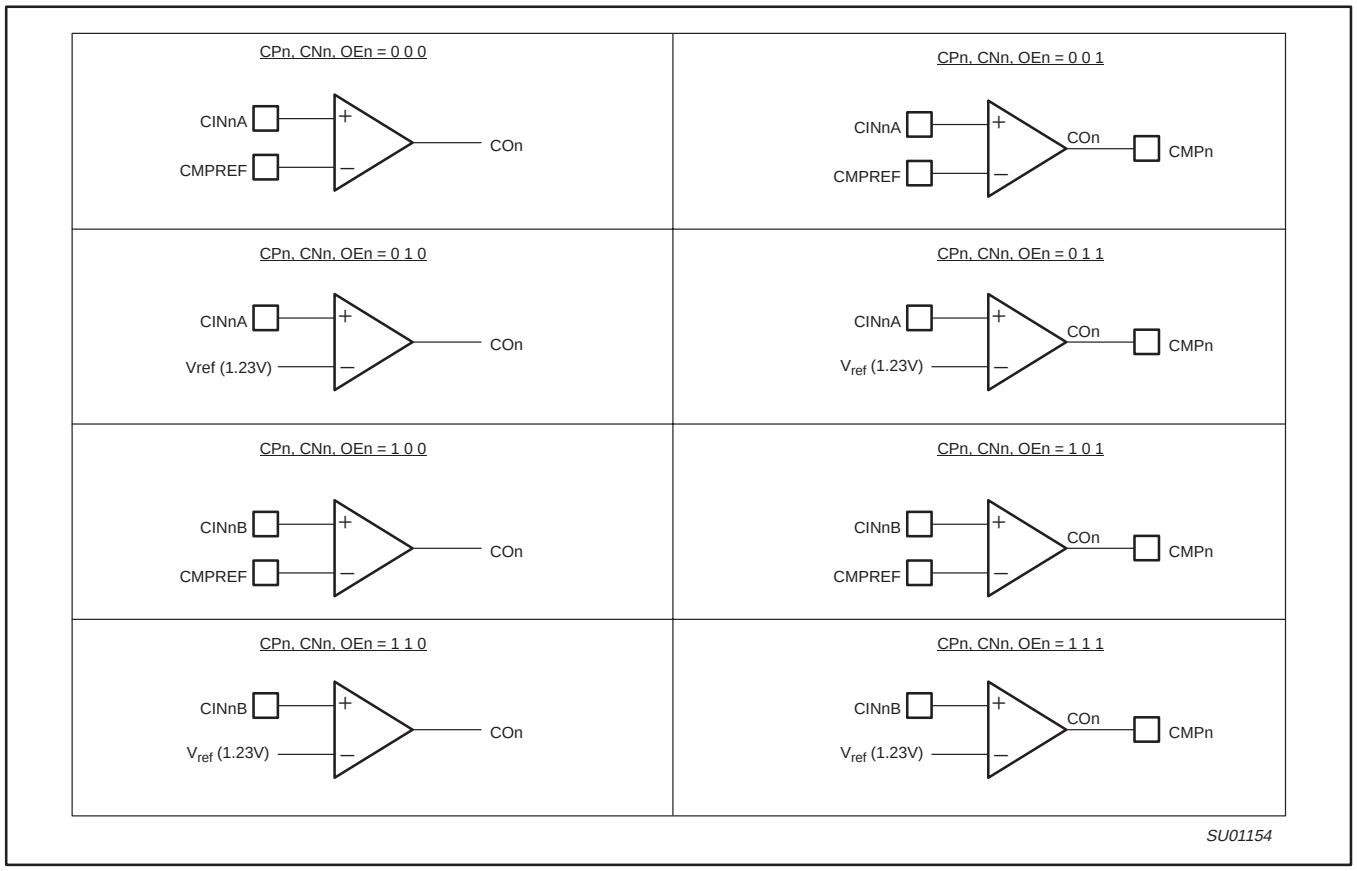


Figure 4. Comparator Configurations

I²C Serial Interface

The I²C bus uses two wires (SDA and SCL) to transfer information between devices connected to the bus. The main features of the bus are:

- Bidirectional data transfer between masters and slaves.
- Serial addressing of slaves (no added wiring).
- Acknowledgment after each transferred byte.
- Multimaster bus.
- Arbitration between simultaneously transmitting masters without corruption of serial data on bus.

The I²C subsystem includes hardware to simplify the software required to drive the I²C bus. The hardware is a single bit interface which in addition to including the necessary arbitration and framing error checks, includes clock stretching and a bus timeout timer. The interface is synchronized to software either through polled loops or interrupts.

Refer to the application note AN422, entitled “Using the 8XC751 Microcontroller as an I²C Bus Master” for additional discussion of the 8xC76x I²C interface and sample driver routines.

The 87LPC762 I²C implementation duplicates that of the 87C751 and 87C752 except for the following details:

- The interrupt vector addresses for both the I²C interrupt and the Timer I interrupt.
- The I²C SFR addresses (I2CON, I2CFG, I2DAT).
- The location of the I²C interrupt enable bit and the name of the SFR it is located within (EI2 is Bit 0 in IEN1).
- The location of the Timer I interrupt enable bit and the name of the SFR it is located within (ETI is Bit 7 in IEN1).
- The I²C and Timer I interrupts have a settable priority.

Timer I is used to both control the timing of the I²C bus and also to detect a “bus locked” condition, by causing an interrupt when nothing happens on the I²C bus for an inordinately long period of time while a transmission is in progress. If this interrupt occurs, the program has the opportunity to attempt to correct the fault and resume I²C operation.

Six time spans are important in I²C operation and are insured by timer I:

- The MINIMUM HIGH time for SCL when this device is the master.
- The MINIMUM LOW time for SCL when this device is a master. This is not very important for a single-bit hardware interface like this one, because the SCL low time is stretched until the software responds to the I²C flags. The software response time normally meets or exceeds the MIN LO time. In cases where the software responds within MIN HI + MIN LO) time, timer I will ensure that the minimum time is met.
- The MINIMUM SCL HIGH TO SDA HIGH time in a stop condition.
- The MINIMUM SDA HIGH TO SDA LOW time between I²C stop and start conditions (4.7ms, see I²C specification).
- The MINIMUM SDA LOW TO SCL LOW time in a start condition.
- The MAXIMUM SCL CHANGE time while an I²C frame is in progress. A frame is in progress between a start condition and the following stop condition. This time span serves to detect a lack of software response on this device as well as external I²C problems. SCL “stuck low” indicates a faulty master or slave. SCL “stuck high” may mean a faulty device, or that noise induced onto the I²C bus caused all masters to withdraw from I²C arbitration.

The first five of these times are 4.7ms (see I²C specification) and are covered by the low order three bits of timer I. Timer I is clocked by the 87LPC762 CPU clock. Timer I can be pre-loaded with one of four values to optimize timing for different oscillator frequencies. At lower frequencies, software response time is increased and will degrade maximum performance of the I²C bus. See special function register I2CFG description for prescale values (CT0, CT1).

The MAXIMUM SCL CHANGE time is important, but its exact span is not critical. The complete 10 bits of timer I are used to count out the maximum time. When I²C operation is enabled, this counter is cleared by transitions on the SCL pin. The timer does not run between I²C frames (i.e., whenever reset or stop occurred more recently than the last start). When this counter is running, it will carry out after 1020 to 1023 machine cycles have elapsed since a change on SCL. A carry out causes a hardware reset of the I²C interface and generates an interrupt if the Timer I interrupt is enabled. In cases where the bus hang-up is due to a lack of software response by this device, the reset releases SCL and allows I²C operation among other devices to continue.

Timer I is enabled to run, and will reset the I²C interface upon overflow, if the TIRUN bit in the I2CFG register is set. The Timer I interrupt may be enabled via the ETI bit in IEN1, and its priority set by the PTIH and PTI bits in the IP1H and IP1 registers respectively.

I²C Interrupts

If I²C interrupts are enabled (EA and EI2 are both set to 1), an I²C interrupt will occur whenever the ATN flag is set by a start, stop, arbitration loss, or data ready condition (refer to the description of ATN following). In practice, it is not efficient to operate the I²C interface in this fashion because the I²C interrupt service routine would somehow have to distinguish between hundreds of possible conditions. Also, since I²C can operate at a fairly high rate, the software may execute faster if the code simply waits for the I²C interface.

Typically, the I²C interrupt should only be used to indicate a start condition at an idle slave device, or a stop condition at an idle master device (if it is waiting to use the I²C bus). This is accomplished by enabling the I²C interrupt only during the aforementioned conditions.

Reading I2CON

RDAT — The data from SDA is captured into “Receive DATa” whenever a rising edge occurs on SCL. RDAT is also available (with seven low-order zeros) in the I2DAT register. The difference between reading it here and there is that reading I2DAT clears DRDY, allowing the I²C to proceed on to another bit. Typically, the first seven bits of a received byte are read from I2DAT, while the 8th is read here. Then I2DAT can be written to send the Acknowledge bit and clear DRDY.

ATN — “ATteNtion” is 1 when one or more of DRDY, ARL, STR, or STP is 1. Thus, ATN comprises a single bit that can be tested to release the I²C service routine from a “wait loop.”

DRDY — “Data ReaDY” (and thus ATN) is set when a rising edge occurs on SCL, except at idle slave. DRDY is cleared by writing CDR = 1, or by writing or reading the I2DAT register. The following low period on SCL is stretched until the program responds by clearing DRDY.

ARL "Arbitration Loss" is 1 when transmit Active was set, but this device lost arbitration to another transmitter. Transmit Active is cleared when ARL is 1. There are four separate cases in which ARL is set.

1. If the program sent a 1 or repeated start, but another device sent a 0, or a stop, so that SDA is 0 at the rising edge of SCL. (If the other device sent a stop, the setting of ARL will be followed shortly by STP being set.)

2. If the program sent a 1, but another device sent a repeated start, and it drove SDA low before SCL could be driven low. (This type of ARL is always accompanied by STR = 1.)

3. In master mode, if the program sent a repeated start, but another device sent a 1, and it drove SCL low before this device could drive SDA low.

4. In master mode, if the program sent stop, but it could not be sent because another device sent a 0.

STR "STaRt" is set to a 1 when an I^2C start condition is detected at a non-idle slave or at a master. (STR is not set when an idle slave becomes active due to a start bit; the slave has nothing useful to do until the rising edge of SCL sets DRDY.)

STP "SToP" is set to 1 when an I^2C stop condition is detected at a non-idle slave or at a master. (STP is not set for a stop condition at an idle slave.)

MASTER "MASTER" is 1 if this device is currently a master on the I^2C. MASTER is set when MASTRQ is 1 and the bus is not busy (i.e., if a start bit hasn't been received since reset or a "Timer I" time-out, or if a stop has been received since the last start). MASTER is cleared when ARL is set, or after the software writes MASTRQ = 0 and then XSTP = 1.

Writing I2CON

Typically, for each bit in an I^2C message, a service routine waits for ATN = 1. Based on DRDY, ARL, STR, and STP, and on the current bit position in the message, it may then write I2CON with one or more of the following bits, or it may read or write the I2DAT register.

CXA Writing a 1 to "Clear Xmit Active" clears the Transmit Active state. (Reading the I2DAT register also does this.)

Regarding Transmit Active

Transmit Active is set by writing the I2DAT register, or by writing I2CON with XSTR = 1 or XSTP = 1. The I^2C interface will only drive the SDA line low when Transmit Active is set, and the ARL bit will only be set to 1 when Transmit Active is set. Transmit Active is cleared by reading the I2DAT register, or by writing I2CON with CXA = 1. Transmit Active is automatically cleared when ARL is 1.

IDLE Writing 1 to "IDLE" causes a slave's I^2C hardware to ignore the I^2C until the next start condition (but if MASTRQ is 1, then a stop condition will cause this device to become a master).

CDR Writing a 1 to "Clear Data Ready" clears DRDY. (Reading or writing the I2DAT register also does this.)

CARL Writing a 1 to "Clear Arbitration Loss" clears the ARL bit.

CSTR Writing a 1 to "Clear STaRt" clears the STR bit.

CSTP Writing a 1 to "Clear SToP" clears the STP bit. Note that if one or more of DRDY, ARL, STR, or STP is 1, the low time of SCL is stretched until the service routine responds by clearing them.

XSTR Writing 1s to "Xmit repeated STaRt" and CDR tells the I^2C hardware to send a repeated start condition. This should only be at a master. Note that XSTR need not and should not be used to send an "initial" (non-repeated) start; it is sent automatically by the I^2C hardware. Writing XSTR = 1 includes the effect of writing I2DAT with XDAT = 1; it sets Transmit Active and releases SDA to high during the SCL low time. After SCL goes high, the I^2C hardware waits for the suitable minimum time and then drives SDA low to make the start condition.

XSTP Writing 1s to "Xmit SToP" and CDR tells the I^2C hardware to send a stop condition. This should only be done at a master. If there are no more messages to initiate, the service routine should clear the MASTRQ bit in I2CFG to 0 before writing XSTP with 1. Writing XSTP = 1 includes the effect of writing I2DAT with XDAT = 0; it sets Transmit Active and drives SDA low during the SCL low time. After SCL goes high, the I^2C hardware waits for the suitable minimum time and then releases SDA to high to make the stop condition.

I2CFG Address: C8h Reset Value: 00h

Bit Addressable

7	6	5	4	3	2	1	0
SLAVEN	MASTRQ	CLRTI	TIRUN	—	—	CT1	CT0

BIT	SYMBOL	FUNCTION
I2CFG.7	SLAVEN	Slave Enable. Writing a 1 this bit enables the slave functions of the I^2C subsystem. If SLAVEN and MASTRQ are 0, the I^2C hardware is disabled. This bit is cleared to 0 by reset and by an I^2C time-out.
I2CFG.6	MASTRQ	Master Request. Writing a 1 to this bit requests mastership of the I^2C bus. If a transmission is in progress when this bit is changed from 0 to 1, action is delayed until a stop condition is detected. A start condition is sent and DRDY is set (thus making ATN = 1 and generating an I^2C interrupt). When a master wishes to release mastership status of the I^2C, it writes a 1 to XSTP in I2CON. MASTRQ is cleared by an I^2C time-out.
I2CFG.5	CLRTI	Writing a 1 to this bit clears the Timer I overflow flag. This bit position always reads as a 0.
I2CFG.4	TIRUN	Writing a 1 to this bit lets Timer I run; a zero stops and clears it. Together with SLAVEN, MASTRQ, and MASTER, this bit determines operational modes as shown in Table 1.
I2CFG.2, 3	—	Reserved for future use. Should not be set to 1 by user programs.
I2CFG.1, 0	CT1, CT0	These two bits are programmed as a function of the CPU clock rate, to optimize the MIN HI and LO time of SCL when this device is a master on the I^2C. The time value determined by these bits controls both of these parameters, and also the timing for stop and start conditions.

SU01157

Figure 8. I^2C Configuration Register (I2CFG)

Regarding Software Response Time

Because the 87LPC762 can run at 20 MHz, and because the I^2C interface is optimized for high-speed operation, it is quite likely that an I^2C service routine will sometimes respond to DRDY (which is set at a rising edge of SCL) and write I2DAT before SCL has gone low again. If XDAT were applied directly to SDA, this situation would produce an I^2C protocol violation. The programmer need not worry about this possibility because XDAT is applied to SDA only when SCL is low.

Conversely, a program that includes an I^2C service routine may take a long time to respond to DRDY. Typically, an I^2C routine operates on a flag-polling basis during a message, with interrupts from other peripheral functions enabled. If an interrupt occurs, it will delay the response of the I^2C service routine. The programmer need not worry about this very much either, because the I^2C hardware stretches the SCL low time until the service routine responds. The only constraint on the response is that it must not exceed the Timer I time-out.

Values to be used in the CT1 and CT0 bits are shown in Table 2. To allow the I^2C bus to run at the maximum rate for a particular oscillator frequency, compare the actual oscillator rate to the f OSC max column in the table. The value for CT1 and CT0 is found in the first line of the table where CPU clock max is greater than or equal to the actual frequency.

Table 2 also shows the machine cycle count for various settings of CT1/CT0. This allows calculation of the actual minimum high and low times for SCL as follows:

$$\text{SCL min high/low time (in microseconds)} = \frac{6 * \text{Min Time Count}}{\text{CPU clock (in MHz)}}$$

For instance, at an 8 MHz frequency, with CT1/CT0 set to 1 0, the minimum SCL high and low times will be 5.25 μs.

Table 2 also shows the Timer I timeout period (given in machine cycles) for each CT1/CT0 combination. The timeout period varies because of the way in which minimum SCL high and low times are measured. When the I^2C interface is operating, Timer I is pre-loaded at every SCL transition with a value dependent upon CT1/CT0. The pre-load value is chosen such that a minimum SCL high or low time has elapsed when Timer I reaches a count of 008 (the actual value pre-loaded into Timer I is 8 minus the machine cycle count).

I/O Ports

The 87LPC762 has 3 I/O ports, port 0, port 1, and port 2. The exact number of I/O pins available depend upon the oscillator and reset options chosen. At least 15 pins of the 87LPC762 may be used as I/Os when a two-pin external oscillator and an external reset circuit are used. Up to 18 pins may be available if fully on-chip oscillator and reset configurations are chosen.

All but three I/O port pins on the 87LPC762 may be software configured to one of four types on a bit-by-bit basis, as shown in Table 4. These are: quasi-bidirectional (standard 80C51 port outputs), push-pull, open drain, and input only. Two configuration registers for each port choose the output type for each port pin.

Table 4. Port Output Configuration Settings

PxM1.y	PxM2.y	Port Output Mode
0	0	Quasi-bidirectional
0	1	Push-Pull
1	0	Input Only (High Impedance)
1	1	Open Drain

Quasi-Bidirectional Output Configuration

The default port output configuration for standard 87LPC762 I/O ports is the quasi-bidirectional output that is common on the 80C51 and most of its derivatives. This output type can be used as both an input and output without the need to reconfigure the port. This is possible because when the port outputs a logic high, it is weakly driven, allowing an external device to pull the pin low. When the pin is pulled low, it is driven strongly and able to sink a fairly large current. These features are somewhat similar to an open drain output except that there are three pull-up transistors in the quasi-bidirectional output that serve different purposes.

One of these pull-ups, called the “very weak” pull-up, is turned on whenever the port latch for the pin contains a logic 1. The very weak pull-up sources a very small current that will pull the pin high if it is left floating.

A second pull-up, called the “weak” pull-up, is turned on when the port latch for the pin contains a logic 1 and the pin itself is also at a logic 1 level. This pull-up provides the primary source current for a quasi-bidirectional pin that is outputting a 1. If a pin that has a logic 1 on it is pulled low by an external device, the weak pull-up turns off, and only the very weak pull-up remains on. In order to pull the pin low under these conditions, the external device has to sink enough current to overpower the weak pull-up and take the voltage on the port pin below its input threshold.

The third pull-up is referred to as the “strong” pull-up. This pull-up is used to speed up low-to-high transitions on a quasi-bidirectional port pin when the port latch changes from a logic 0 to a logic 1. When this occurs, the strong pull-up turns on for a brief time, two CPU clocks, in order to pull the port pin high quickly. Then it turns off again.

The quasi-bidirectional port configuration is shown in Figure 10.

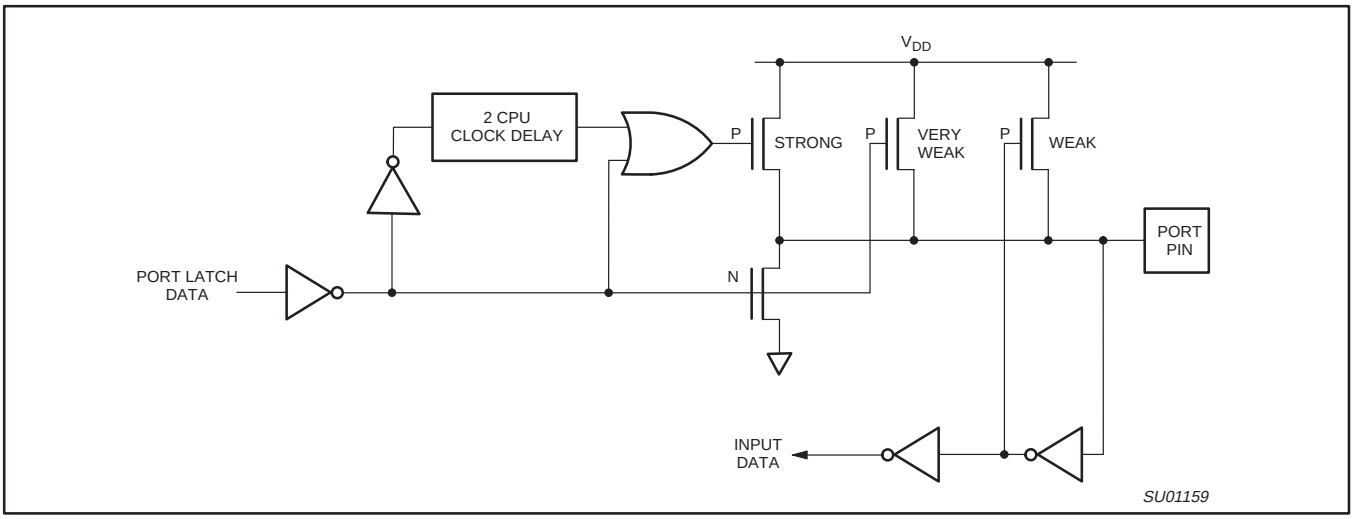


Figure 10. Quasi-Bidirectional Output

Open Drain Output Configuration

The open drain output configuration turns off all pull-ups and only drives the pull-down transistor of the port driver when the port latch contains a logic 0. To be used as a logic output, a port configured in this manner must have an external pull-up, typically a resistor tied to V_{DD}. The pull-down for this mode is the same as for the quasi-bidirectional mode.

The open drain port configuration is shown in Figure 11.

Push-Pull Output Configuration

The push-pull output configuration has the same pull-down structure as both the open drain and the quasi-bidirectional output modes, but provides a continuous strong pull-up when the port latch contains a logic 1. The push-pull mode may be used when more source current is needed from a port output.

The push-pull port configuration is shown in Figure 12.

The three port pins that cannot be configured are P1.2, P1.3, and P1.5. The port pins P1.2 and P1.3 are permanently configured as open drain outputs. They may be used as inputs by writing ones to their respective port latches. P1.5 may be used as a Schmitt trigger input if the 87LPC762 has been configured for an internal reset and is not using the external reset input function $\overline{RST}$.

Additionally, port pins P2.0 and P2.1 are disabled for both input and output if one of the crystal oscillator options is chosen. Those options are described in the Oscillator section.

The value of port pins at reset is determined by the PRHI bit in the UCFG1 register. Ports may be configured to reset high or low as needed for the application. When port pins are driven high at reset, they are in quasi-bidirectional mode and therefore do not source large amounts of current.

Every output on the 87LPC762 may potentially be used as a 20 mA sink LED drive output. However, there is a maximum total output current for all ports which must not be exceeded.

All ports pins of the 87LPC762 have slew rate controlled outputs. This is to limit noise generated by quickly switching output signals. The slew rate is factory set to approximately 10 ns rise and fall times.

The bits in the P2M1 register that are not used to control configuration of P2.1 and P2.0 are used for other purposes. These bits can enable Schmitt trigger inputs on each I/O port, enable toggle outputs from Timer 0 and Timer 1, and enable a clock output if either the internal RC oscillator or external clock input is being used. The last two functions are described in the Timer/Counters and Oscillator sections respectively. The enable bits for all of these functions are shown in Figure 13.

Each I/O port of the 87LPC762 may be selected to use TTL level inputs or Schmitt inputs with hysteresis. A single configuration bit determines this selection for the entire port. Port pins P1.2, P1.3, and P1.5 always have a Schmitt trigger input.

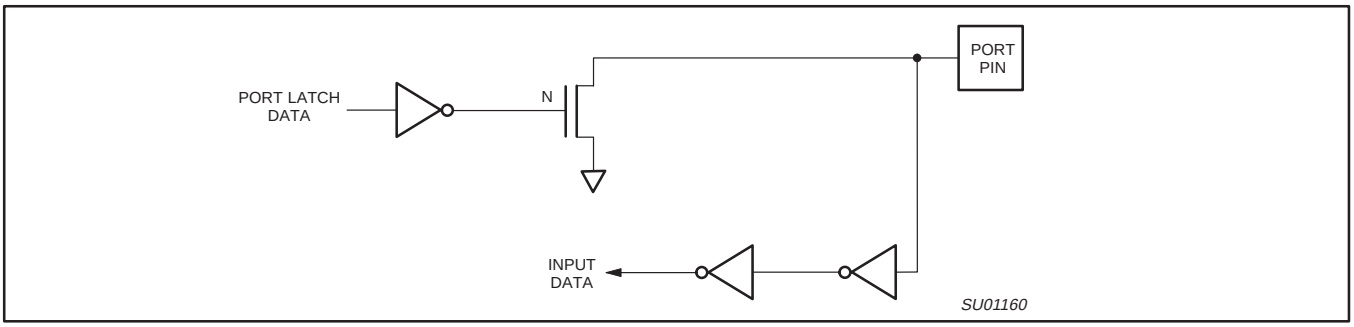


Figure 11. Open Drain Output

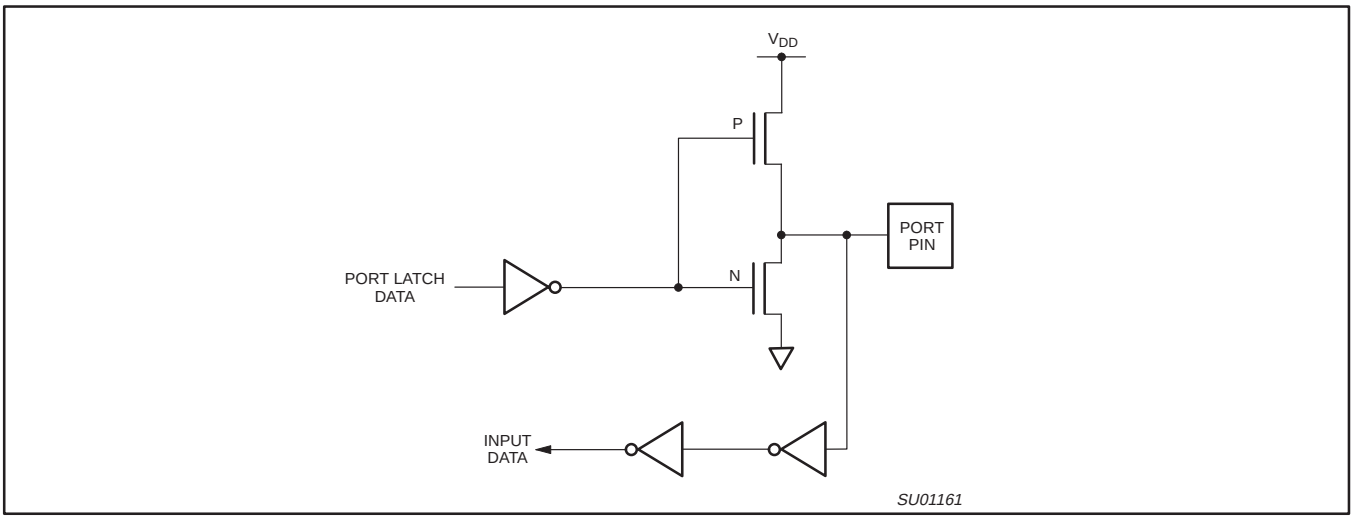


Figure 12. Push-Pull Output

For correct activation of Brownout Detect, the V_{DD} fall time must be no faster than 50 mV/µs. When V_{DD} is restored, is should not rise faster than 2 mV/µs in order to insure a proper reset.

The brownout voltage (2.5 V or 3.8 V) is selected via the BOV bit in the EPROM configuration register UCFG1. When unprogrammed (BOV = 1), the brownout detect voltage is 2.5 V. When programmed (BOV = 0), the brownout detect voltage is 3.8 V.

If the Brownout Detect function is not required in an application, it may be disabled, thus saving power. Brownout Detect is disabled by setting the control bit BOD in the AUXR1 register (AUXR1.6).

Power On Detection

The Power On Detect has a function similar to the Brownout Detect, but is designed to work as power comes up initially, before the power supply voltage reaches a level where Brownout Detect can work. When this feature is activated, the POF flag in the PCON register is set to indicate an initial power up condition. The POF flag will remain set until cleared by software.

Power Reduction Modes

The 87LPC762 supports Idle and Power Down modes of power reduction.

Idle Mode

The Idle mode leaves peripherals running in order to allow them to activate the processor when an interrupt is generated. Any enabled interrupt source or Reset may terminate Idle mode. Idle mode is entered by setting the IDL bit in the PCON register (see Figure 19).

Power Down Mode

The Power Down mode stops the oscillator in order to absolutely minimize power consumption. Power Down mode is entered by setting the PD bit in the PCON register (see Figure 19).

The processor can be made to exit Power Down mode via Reset or one of the interrupt sources shown in Table 8. This will occur if the interrupt is enabled and its priority is higher than any interrupt currently in progress.

In Power Down mode, the power supply voltage may be reduced to the RAM keep-alive voltage V_{RAM}. This retains the RAM contents at the point where Power Down mode was entered. SFR contents are not guaranteed after V_{DD} has been lowered to V_{RAM}, therefore it is recommended to wake up the processor via Reset in this case. V_{DD} must be raised to within the operating range before the Power Down mode is exited. Since the watchdog timer has a separate oscillator, it may reset the processor upon overflow if it is running during Power Down.

Note that if the Brownout Detect reset is enabled, the processor will be put into reset as soon as V_{DD} drops below the brownout voltage. If Brownout Detect is configured as an interrupt and is enabled, it will wake up the processor from Power Down mode when V_{DD} drops below the brownout voltage.

When the processor wakes up from Power Down mode, it will start the oscillator immediately and begin execution when the oscillator is stable. Oscillator stability is determined by counting 1024 CPU clocks after start-up when one of the crystal oscillator configurations is used, or 256 clocks after start-up for the internal RC or external clock input configurations.

Some chip functions continue to operate and draw power during Power Down mode, increasing the total power used during Power Down. These include the Brownout Detect, Watchdog Timer, and Comparators.

PCON Address: 87h

Not Bit Addressable

Reset Value:
- 30h for a Power On reset
- 20h for a Brownout reset
- 00h for other reset sources

7	6	5	4	3	2	1	0
SMOD1	SMOD0	BOF	POF	GF1	GF0	PD	IDL

BIT	SYMBOL	FUNCTION
PCON.7	SMOD1	When set, this bit doubles the UART baud rate for modes 1, 2, and 3.
PCON.6	SMOD0	This bit selects the function of bit 7 of the SCON SFR. When 0, SCON.7 is the SM0 bit. When 1, SCON.7 is the FE (Framing Error) flag. See Figure 26 for additional information.
PCON.5	BOF	Brown Out Flag. Set automatically when a brownout reset or interrupt has occurred. Also set at power on. Cleared by software. Refer to the Power Monitoring Functions section for additional information.
PCON.4	POF	Power On Flag. Set automatically when a power-on reset has occurred. Cleared by software. Refer to the Power Monitoring Functions section for additional information.
PCON.3	GF1	General purpose flag 1. May be read or written by user software, but has no effect on operation.
PCON.2	GF0	General purpose flag 0. May be read or written by user software, but has no effect on operation.
PCON.1	PD	Power Down control bit. Setting this bit activates Power Down mode operation. Cleared when the Power Down mode is terminated (see text).
PCON.0	IDL	Idle mode control bit. Setting this bit activates Idle mode operation. Cleared when the Idle mode is terminated (see text).

SU01168

Figure 19. Power Control Register (PCON)

Table 8. Sources of Wakeup from Power Down Mode

Wakeup Source	Conditions
External Interrupt 0 or 1	The corresponding interrupt must be enabled.
Keyboard Interrupt	The keyboard interrupt feature must be enabled and properly set up. The corresponding interrupt must be enabled.
Comparator 1 or 2	The comparator(s) must be enabled and properly set up. The corresponding interrupt must be enabled.
Watchdog Timer Reset	The watchdog timer must be enabled via the WDTE bit in the UCFG1 EPROM configuration byte.
Watchdog Timer Interrupt	The WDTE bit in the UCFG1 EPROM configuration byte must not be set. The corresponding interrupt must be enabled.
Brownout Detect Reset	The BOD bit in AUXR1 must not be set (brownout detect not disabled). The BOI bit in AUXR1 must not be set (brownout interrupt disabled).
Brownout Detect Interrupt	The BOD bit in AUXR1 must not be set (brownout detect not disabled). The BOI bit in AUXR1 must be set (brownout interrupt enabled). The corresponding interrupt must be enabled.
Reset Input	The external reset input must be enabled.

Timer/Counters

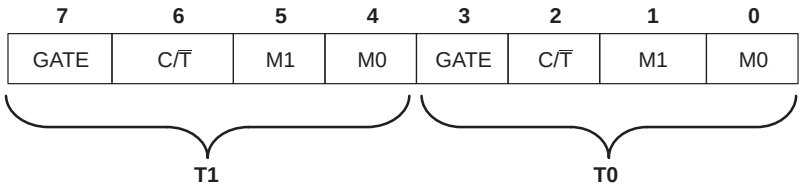
The 87LPC762 has two general purpose counter/timers which are upward compatible with the standard 80C51 Timer 0 and Timer 1. Both can be configured to operate either as timers or event counters (see Figure 22). An option to automatically toggle the T0 and/or T1 pins upon timer overflow has been added.

In the “Timer” function, the register is incremented every machine cycle. Thus, one can think of it as counting machine cycles. Since a machine cycle consists of 6 CPU clock periods, the count rate is 1/6 of the CPU clock frequency. Refer to the section Enhanced CPU for a description of the CPU clock.

In the “Counter” function, the register is incremented in response to a 1-to-0 transition at its corresponding external input pin, T0 or T1. In this function, the external input is sampled once during every machine cycle. When the samples of the pin state show a high in one cycle and a low in the next cycle, the count is incremented. The new count value appears in the register during the cycle following the one in which the transition was detected. Since it takes 2 machine cycles (12 CPU clocks) to recognize a 1-to-0 transition, the maximum count rate is 1/6 of the CPU clock frequency. There are no restrictions on the duty cycle of the external input signal, but to ensure that a given level is sampled at least once before it changes, it should be held for at least one full machine cycle.

The “Timer” or “Counter” function is selected by control bits C/$\overline{T}$ in the Special Function Register TMOD. In addition to the “Timer” or “Counter” selection, Timer 0 and Timer 1 have four operating modes, which are selected by bit-pairs (M1, M0) in TMOD. Modes 0, 1, and 2 are the same for both Timers/Counters. Mode 3 is different. The four operating modes are described in the following text.

TMOD Address: 89h Reset Value: 00h
Not Bit Addressable



BIT	SYMBOL	FUNCTION
TMOD.7	GATE	Gating control for Timer 1. When set, Timer/Counter is enabled only while the $\overline{INT1}$ pin is high and the TR1 control pin is set. When cleared, Timer 1 is enabled when the TR1 control bit is set.
TMOD.6	C/$\overline{T}$	Timer or Counter Selector for Timer 1. Cleared for Timer operation (input from internal system clock.) Set for Counter operation (input from T1 input pin).
TMOD.5, 4	M1, M0	Mode Select for Timer 1 (see table below).
TMOD.3	GATE	Gating control for Timer 0. When set, Timer/Counter is enabled only while the $\overline{INT0}$ pin is high and the TR0 control pin is set. When cleared, Timer 0 is enabled when the TR0 control bit is set.
TMOD.2	C/$\overline{T}$	Timer or Counter Selector for Timer 0. Cleared for Timer operation (input from internal system clock.) Set for Counter operation (input from T0 input pin).
TMOD.1, 0	M1, M0	Mode Select for Timer 0 (see table below).
	M1, M0	Timer Mode
	0 0	8048 Timer “TLn” serves as 5-bit prescaler.
	0 1	16-bit Timer/Counter “THn” and “TLn” are cascaded; there is no prescaler.
	1 0	8-bit auto-reload Timer/Counter. THn holds a value which is loaded into TLn when it overflows.
	1 1	Timer 0 is a dual 8-bit Timer/Counter in this mode. TL0 is an 8-bit Timer/Counter controlled by the standard Timer 0 control bits. TH0 is an 8-bit timer only, controlled by the Timer 1 control bits (see text). Timer 1 in this mode is stopped.

SU01171

Figure 22. Timer/Counter Mode Control Register (TMOD)

Mode 1
Mode 1 is the same as Mode 0, except that all 16 bits of the timer register (THn and TLn) are used. See Figure 25

Mode 2
Mode 2 configures the Timer register as an 8-bit Counter (TL1) with automatic reload, as shown in Figure 26. Overflow from TLn not only sets TFn, but also reloads TLn with the contents of THn, which must be preset by software. The reload leaves THn unchanged. Mode 2 operation is the same for Timer 0 and Timer 1.

Mode 3
When Timer 1 is in Mode 3 it is stopped. The effect is the same as setting TR1 = 0.

Timer 0 in Mode 3 establishes TL0 and TH0 as two separate 8-bit counters. The logic for Mode 3 on Timer 0 is shown in Figure 27. TL0 uses the Timer 0 control bits: C/$\overline{T}$, GATE, TR0 and pin $\overline{INT0}$, and TF0. TH0 is locked into a timer function (counting machine cycles) and takes over the use of TR1 and TF1 from Timer 1. Thus, TH0 now controls the “Timer 1” interrupt.

Mode 3 is provided for applications that require an extra 8-bit timer. With Timer 0 in Mode 3, an 87LPC762 can look like it has three Timer/Counters. When Timer 0 is in Mode 3, Timer 1 can be turned on and off by switching it into and out of its own Mode 3. It can still be used by the serial port as a baud rate generator, or in any application not requiring an interrupt.

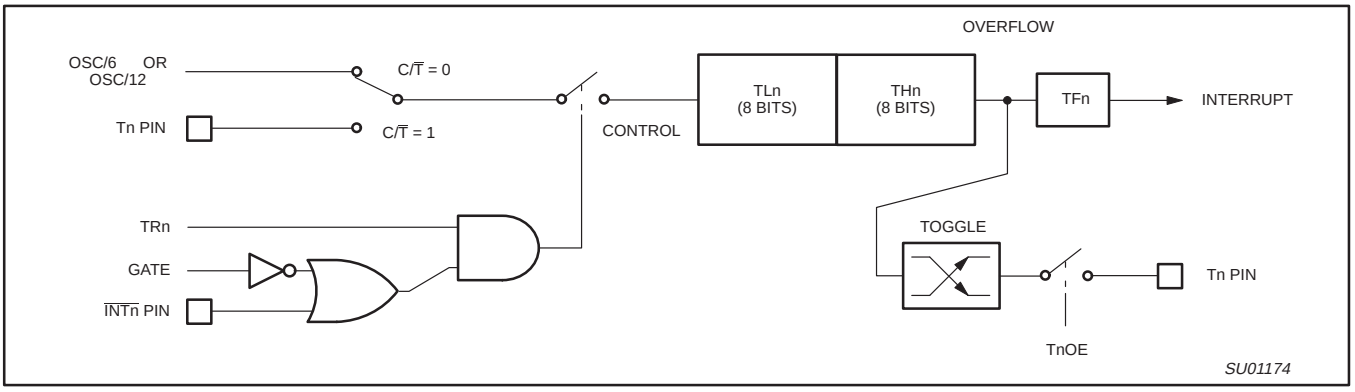


Figure 25. Timer/Counter 0 or 1 in Mode 1 (16-Bit Counter)

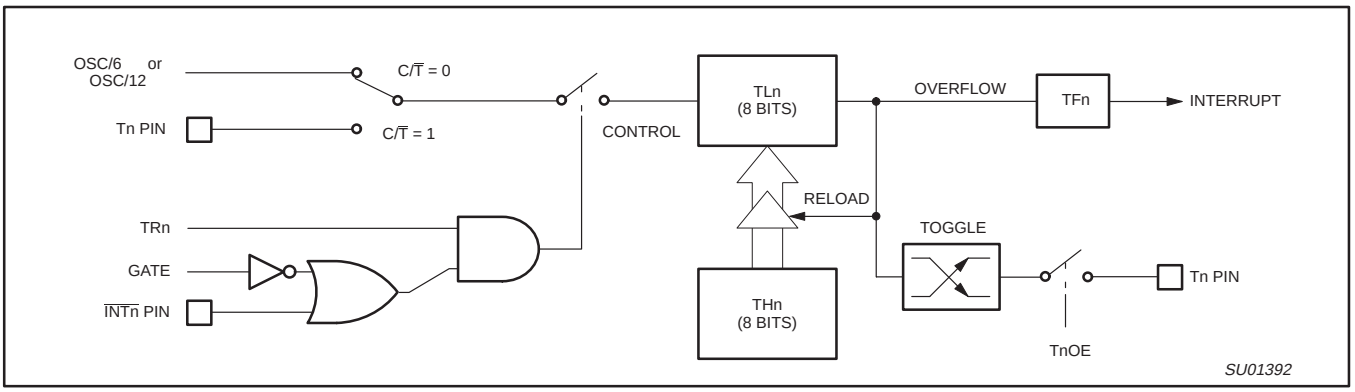


Figure 26. Timer/Counter 0 or 1 in Mode 2 (8-Bit Auto-Reload)

Baud Rates

The baud rate in Mode 0 is fixed: Mode 0 Baud Rate = CPU clock/6. The baud rate in Mode 2 depends on the value of bit SMOD1 in Special Function Register PCON. If SMOD1 = 0 (which is the value on reset), the baud rate is 1/32 of the CPU clock frequency. If SMOD1 = 1, the baud rate is 1/16 of the CPU clock frequency.

$$\text{Mode 2 Baud Rate} = \frac{1 + \text{SMOD1}}{32} \times \text{CPU clock frequency}$$

Using Timer 1 to Generate Baud Rates

When Timer 1 is used as the baud rate generator, the baud rates in Modes 1 and 3 are determined by the Timer 1 overflow rate and the value of SMOD1. The Timer 1 interrupt should be disabled in this application. The Timer itself can be configured for either “timer” or “counter” operation, and in any of its 3 running modes. In the most typical applications, it is configured for “timer” operation, in the auto-reload mode (high nibble of TMOD = 0010b). In that case the baud rate is given by the formula:

$$\text{Mode 1, 3 Baud Rate} = \frac{\text{CPU clock frequency}/\text{192 (or 96 if SMOD1 = 1)}}{256 - \text{(TH1)}}$$

Tables 6 and 7 list various commonly used baud rates and how they can be obtained using Timer 1 as the baud rate generator.

Table 9. Baud Rates, Timer Values, and CPU Clock Frequencies for SMOD1 = 0

Timer Count	Baud Rate					
	2400	4800	9600	19.2k	38.4k	57.6k
–1	0.4608	0.9216	* 1.8432	* 3.6864	* 7.3728	* 11.0592
–2	0.9216	1.8432	* 3.6864	* 7.3728	* 14.7456	
–3	1.3824	2.7648	5.5296	* 11.0592	–	–
–4	* 1.8432	* 3.6864	* 7.3728	* 14.7456	–	–
–5	2.3040	4.6080	9.2160	* 18.4320	–	–
–6	2.7648	5.5296	* 11.0592	–	–	–
–7	3.2256	6.4512	12.9024	–	–	–
–8	* 3.6864	* 7.3728	* 14.7456	–	–	–
–9	4.1472	8.2944	16.5888	–	–	–
–10	4.6080	9.2160	* 18.4320	–	–	–

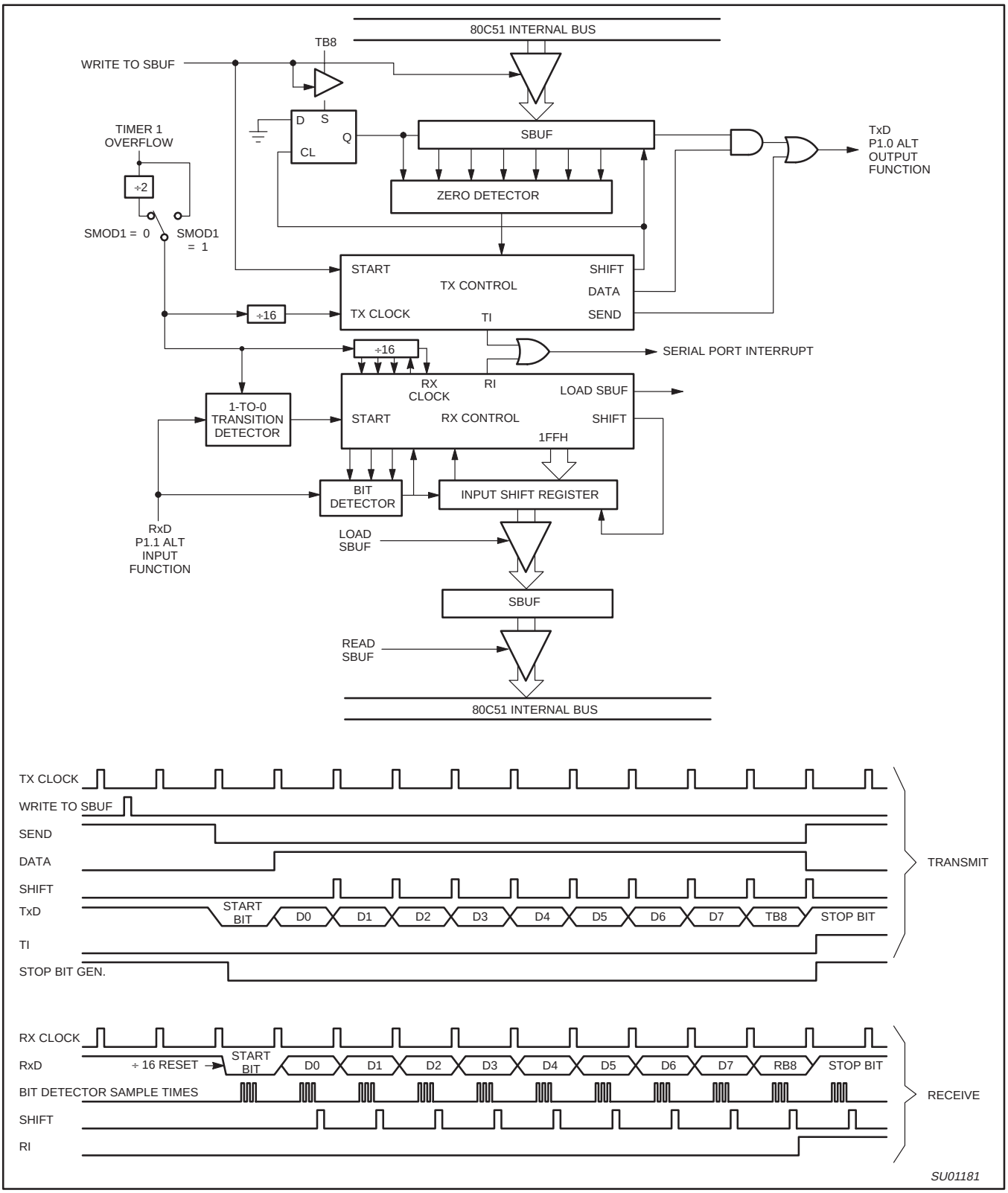


Figure 32. Serial Port Mode 3

Automatic Address Recognition

Automatic Address Recognition is a feature which allows the UART to recognize certain addresses in the serial bit stream by using hardware to make the comparisons. This feature saves a great deal of software overhead by eliminating the need for the software to examine every serial address which passes by the serial port. This feature is enabled by setting the SM2 bit in SCON. In the 9 bit UART modes, mode 2 and mode 3, the Receive Interrupt flag (RI) will be automatically set when the received byte contains either the “Given” address or the “Broadcast” address. The 9 bit mode requires that the 9th information bit is a 1 to indicate that the received information is an address and not data.

Using the Automatic Address Recognition feature allows a master to selectively communicate with one or more slaves by invoking the Given slave address or addresses. All of the slaves may be contacted by using the Broadcast address. Two special Function Registers are used to define the slave’s address, SADDR, and the address mask, SADEN. SADEN is used to define which bits in the SADDR are to be used and which bits are “don’t care”. The SADEN mask can be logically ANDed with the SADDR to create the “Given” address which the master will use for addressing each of the slaves. Use of the Given address allows multiple slaves to be recognized while excluding others. The following examples will help to show the versatility of this scheme:

Slave 0 SADDR = 1100 0000
SADEN = 1111 1101
Given = 1100 00X0

Slave 1 SADDR = 1100 0000
SADEN = 1111 1110
Given = 1100 000X

In the above example SADDR is the same and the SADEN data is used to differentiate between the two slaves. Slave 0 requires a 0 in bit 0 and it ignores bit 1. Slave 1 requires a 0 in bit 1 and bit 0 is ignored. A unique address for Slave 0 would be 1100 0010 since slave 1 requires a 0 in bit 1. A unique address for slave 1 would be 1100 0001 since a 1 in bit 0 will exclude slave 0. Both slaves can be selected at the same time by an address which has bit 0 = 0 (for slave 0) and bit 1 = 0 (for slave 1). Thus, both could be addressed with 1100 0000.

In a more complex system the following could be used to select slaves 1 and 2 while excluding slave 0:

Slave 0 SADDR = 1100 0000
SADEN = 1111 1001
Given = 1100 0XX0

Slave 1 SADDR = 1110 0000
SADEN = 1111 1010
Given = 1110 0X0X

Slave 2 SADDR = 1110 0000
SADEN = 1111 1100
Given = 1110 00XX

In the above example the differentiation among the 3 slaves is in the lower 3 address bits. Slave 0 requires that bit 0 = 0 and it can be uniquely addressed by 1110 0110. Slave 1 requires that bit 1 = 0 and it can be uniquely addressed by 1110 and 0101. Slave 2 requires that bit 2 = 0 and its unique address is 1110 0011. To select Slaves 0 and 1 and exclude Slave 2 use address 1110 0100, since it is necessary to make bit 2 = 1 to exclude slave 2. The Broadcast Address for each slave is created by taking the logical OR of SADDR and SADEN. Zeros in this result are treated as don’t-cares. In most cases, interpreting the don’t-cares as ones, the broadcast address will be FF hexadecimal. Upon reset SADDR and SADEN are loaded with 0s. This produces a given address of all “don’t cares” as well as a Broadcast address of all “don’t cares”. This effectively disables the Automatic Addressing mode and allows the microcontroller to use standard UART drivers which do not make use of this feature.

Watchdog Timer

When enabled via the WDTE configuration bit, the watchdog timer is operated from an independent, fully on-chip oscillator in order to provide the greatest possible dependability. When the watchdog feature is enabled, the timer must be fed regularly by software in order to prevent it from resetting the CPU, and it **cannot** be turned off. When disabled as a watchdog timer (via the WDTE bit in the UCFG1 configuration register), it may be used as an interval timer and may generate an interrupt. The watchdog timer is shown in Figure 33.

The watchdog timeout time is selectable from one of eight values, nominal times range from 25 milliseconds to 3.2 seconds. The frequency tolerance of the independent watchdog RC oscillator is ±60%. The timeout selections and other control bits are shown in Figure 34. When the watchdog function is enabled, the WDCON register may be written once during chip initialization in order to set the watchdog timeout time. The recommended method of initializing the WDCON register is to first feed the watchdog, then write to WDCON to configure the WDS2–0 bits. Using this method, the watchdog initialization may be done any time within 10 milliseconds after startup without a watchdog overflow occurring before the initialization can be completed.

Since the watchdog timer oscillator is fully on-chip and independent of any external oscillator circuit used by the CPU, it intrinsically serves as an oscillator fail detection function. If the watchdog feature is enabled and the CPU oscillator fails for any reason, the watchdog timer will time out and reset the CPU.

When the watchdog function is enabled, the timer is deactivated temporarily when a chip reset occurs from another source, such as a power on reset, brownout reset, or external reset.

Watchdog Feed Sequence

If the watchdog timer is running, it must be fed before it times out in order to prevent a chip reset from occurring. The watchdog feed sequence consists of first writing the value 1Eh, then the value E1h to the WDRST register. An example of a watchdog feed sequence is shown below.

```
WDFeed:
   mov   WDRST,#1eh   ; First part of watchdog feed sequence.
   mov   WDRST,#0e1h  ; Second part of watchdog feed sequence.
```

The two writes to WDRST do not have to occur in consecutive instructions. An incorrect watchdog feed sequence does not cause any immediate response from the watchdog timer, which will still time out at the originally scheduled time if a correct feed sequence does not occur prior to that time.

After a chip reset, the user program has a limited time in which to either feed the watchdog timer or change the timeout period. When a low CPU clock frequency is used in the application, the number of instructions that can be executed before the watchdog overflows may be quite small.

Watchdog Reset

If a watchdog reset occurs, the internal reset is active for approximately one microsecond. If the CPU clock was still running, code execution will begin immediately after that. If the processor was in Power Down mode, the watchdog reset will start the oscillator and code execution will resume after the oscillator is stable.

Additional Features

The AUXR1 register contains several special purpose control bits that relate to several chip features. AUXR1 is described in Figure 35.

Software Reset

The SRST bit in AUXR1 allows software the opportunity to reset the processor completely, as if an external reset or watchdog reset had occurred. If a value is written to AUXR1 that contains a 1 at bit position 3, all SFRs will be initialized and execution will resume at program address 0000. Care should be taken when writing to AUXR1 to avoid accidental software resets.

Dual Data Pointers

The dual Data Pointer (DPTR) adds to the ways in which the processor can specify the address used with certain instructions. The DPS bit in the AUXR1 register selects one of the two Data Pointers. The DPTR that is not currently selected is not accessible to software unless the DPS bit is toggled.

Specific instructions affected by the Data Pointer selection are:

- INC DPTR — Increments the Data Pointer by 1.
- JMP @A+DPTR — Jump indirect relative to DPTR value.
- MOV DPTR, #data16 — Load the Data Pointer with a 16-bit constant.
- MOVC A, @A+DPTR — Move code byte relative to DPTR to the accumulator.
- MOVX A, @DPTR — Move data byte the accumulator to data memory relative to DPTR.
- MOVX @DPTR, A — Move data byte from data memory relative to DPTR to the accumulator.

Also, any instruction that reads or manipulates the DPH and DPL registers (the upper and lower bytes of the current DPTR) will be affected by the setting of DPS. The MOVX instructions have limited application for the 87LPC762 since the part does not have an external data bus. However, they may be used to access EPROM configuration information (see EPROM Characteristics section).

Bit 2 of AUXR1 is permanently wired as a logic 0. This is so that the DPS bit may be toggled (thereby switching Data Pointers) simply by incrementing the AUXR1 register, without the possibility of inadvertently altering other bits in the register.

AUXR1 Address: A2h — Reset Value: 00h

Not Bit Addressable

7	6	5	4	3	2	1	0
KBF	BOD	BOI	LPEP	SRST	0	—	DPS

BIT	SYMBOL	FUNCTION
AUXR1.7	KBF	Keyboard Interrupt Flag. Set when any pin of port 0 that is enabled for the Keyboard Interrupt function goes low. Must be cleared by software.
AUXR1.6	BOD	Brown Out Disable. When set, turns off brownout detection and saves power. See Power Monitoring Functions section for details.
AUXR1.5	BOI	Brown Out Interrupt. When set, prevents brownout detection from causing a chip reset and allows the brownout detect function to be used as an interrupt. See the Power Monitoring Functions section for details.
AUXR1.4	LPEP	Low Power EPROM control bit. Allows power savings in low voltage systems. Set by software. Can only be cleared by power-on or brownout reset. See the Power Reduction Modes section for details.
AUXR1.3	SRST	Software Reset. When set by software, resets the 87LPC762 as if a hardware reset occurred.
AUXR1.2	—	This bit contains a hard-wired 0. Allows toggling of the DPS bit by incrementing AUXR1, without interfering with other bits in the register.
AUXR1.1	—	Reserved for future use. Should not be set to 1 by user programs.
AUXR1.0	DPS	Data Pointer Select. Chooses one of two Data Pointers for use by the program. See text for details.

SU01223

Figure 35. AUXR1 Register

EPROM Characteristics

Programming of the EPROM on the 87LPC762 is accomplished with a serial programming method. Commands, addresses, and data are transmitted to and from the device on two pins after programming mode is entered. Serial programming allows easy implementation of in-circuit programming of the 87LPC762 in an application board.

The 87LPC762 contains three signature bytes that can be read and used by an EPROM programming system to identify the device. The signature bytes designate the device as an 87LPC762 manufactured by Philips. The signature bytes may be read by the user program at addresses FC30h, FC31h and FC60h with the MOVC instruction, using the DPTR register for addressing.

A special user data area is also available for access via the MOVC instruction at addresses FCE0h through FCFFh. This “customer code” space is programmed in the same manner as the main code EPROM and may be used to store a serial number, manufacturing date, or other application information.

32-Byte Customer Code Space

A small supplemental EPROM space is reserved for use by the customer in order to identify code revisions, store checksums, add a serial number to each device, or any other desired use. This area exists in the code memory space from addresses FCE0h through FCFFh. Code execution from this space is not supported, but it may be read as data through the use of the MOVC instruction with the appropriate addresses. The memory may be programmed at the same time as the rest of the code memory and UCFG bytes are programmed.

System Configuration Bytes

A number of user configurable features of the 87LPC762 must be defined at power up and therefore cannot be set by the program after start of execution. Those features are configured through the use of two EPROM bytes that are programmed in the same manner as the EPROM program space. The contents of the two configuration bytes, UCFG1 and UCFG2, are shown in Figures 36 and 37. The values of these bytes may be read by the program through the use of the MOVX instruction at the addresses shown in the figure.

UCFG1 Address: FD00h Unprogrammed Value: FFh

7	6	5	4	3	2	1	0
WDTE	RPD	PRHI	BOV	CLKR	FOSC2	FOSC1	FOSC0

BIT	SYMBOL	FUNCTION
UCFG1.7	WDTE	Watchdog timer enable. When programmed (0), disables the watchdog timer. The timer may still be used to generate an interrupt.
UCFG1.6	RPD	Reset pin disable. When 1 disables the reset function of pin P1.5, allowing it to be used as an input only port pin.
UCFG1.5	PRHI	Port reset high. When 1, ports reset to a high state. When 0, ports reset to a low state.
UCFG1.4	BOV	Brownout voltage select. When 1, the brownout detect voltage is 2.5V. When 0, the brownout detect voltage is 3.8V. This is described in the Power Monitoring Functions section.
UCFG1.3	CLKR	Clock rate select. When 0, the CPU clock rate is divided by 2. This results in machine cycles taking 12 CPU clocks to complete as in the standard 80C51. For full backward compatibility, this division applies to peripheral timing as well.
UCFG1.2–0	FOSC2–FSOC0	CPU oscillator type select. See Oscillator section for additional information. Combinations other than those shown below should not be used. They are reserved for future use.
	FOSC2–FOSC0	Oscillator Configuration
	1 1 1	External clock input on X1 (default setting for an unprogrammed part).
	0 1 1	Internal RC oscillator, 6 MHz. For tolerance, see AC Electrical Characteristics table.
	0 1 0	Low frequency crystal, 20 kHz to 100 kHz.
	0 0 1	Medium frequency crystal or resonator, 100 kHz to 4 MHz.
	0 0 0	High frequency crystal or resonator, 4 MHz to 20 MHz.

SU01477

Figure 36. EPROM System Configuration Byte 1 (UCFG1)

UCFG2 Address: FD01h Unprogrammed Value: FFh

7	6	5	4	3	2	1	0
SB2	SB1	—	—	—	—	—	—

BIT	SYMBOL	FUNCTION
UCFG2.7, 6	SB2, SB1	EPROM security bits. See table entitled, “EPROM Security Bits” for details.
UCFG2.5–0	—	Reserved for future use.

SU01186

Figure 37. EPROM System Configuration Byte 2 (UCFG2)

Security Bits
When neither of the security bits are programmed, the code in the EPROM can be verified. When only security bit 1 is programmed, all further programming of the EPROM is disabled. At that point, only security bit 2 may still be programmed. When both security bits are programmed, EPROM verify is also disabled.

Table 11. EPROM Security Bits

SB2	SB1	Protection Description
1	1	Both security bits unprogrammed. No program security features enabled. EPROM is programmable and verifiable.
1	0	Only security bit 1 programmed. Further EPROM programming is disabled. Security bit 2 may still be programmed.
0	1	Only security bit 2 programmed. This combination is not supported.
0	0	Both security bits programmed. All EPROM verification and programming are disabled.

ABSOLUTE MAXIMUM RATINGS

PARAMETER	RATING	UNIT
Operating temperature under bias	–55 to +125	°C
Storage temperature range	–65 to +150	°C
Voltage on $\overline{RST}/V_{PP}$ pin to V_{SS}	0 to +11.0	V
Voltage on any other pin to V_{SS}	–0.5 to V_{DD}+0.5V	V
Maximum I_{OL} per I/O pin	20	mA
Power dissipation (based on package heat transfer, not device power consumption)	1.5	W

NOTES:
1. Stresses above those listed under Absolute Maximum Ratings may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any conditions other than those described in the AC and DC Electrical Characteristics section of this specification are not implied.
2. This product includes circuitry specifically designed for the protection of its internal devices from the damaging effects of excessive static charge. Nonetheless, it is suggested that conventional precautions be taken to avoid applying greater than the rated maximum.
3. Parameters are valid over operating temperature range unless otherwise specified. All voltages are with respect to VSS unless otherwise noted.

DC ELECTRICAL CHARACTERISTICS

V_{DD} = 2.7 V to 6.0 V unless otherwise specified; T_{amb} = 0°C to +70°C or –40°C to +85°C, unless otherwise specified.

SYMBOL	PARAMETER	TEST CONDITIONS	LIMITS			UNIT
			MIN	TYP[1]	MAX	
I_{DD}	Power supply current, operating	5.0 V, 20 MHz[11]		15	25	mA
		3.0 V, 10 MHz[11]		4	7	mA
I_{ID}	Power supply current, Idle mode	5.0 V, 20 MHz[11]		6	10	mA
		3.0 V, 10 MHz[11]		2	4	mA
I_{PD}	Power supply current, Power Down mode	5.0 V[11]		1	10	µA
		3.0 V[11]		1	5	µA
V_{RAM}	RAM keep-alive voltage		1.5			V
V_{IL}	Input low voltage (TTL input)	4.0 V < V_{DD} < 6.0 V	–0.5		0.2 V_{DD}–0.1	V
		2.7 V < V_{DD} < 4.0 V	–0.5		0.7	V
V_{IL1}	Negative going threshold (Schmitt input)		–0.5		0.3 V_{DD}	V
V_{IH}	Input high voltage (TTL input)		0.2 V_{DD}+0.9		V_{DD}+0.5	V
V_{IH1}	Positive going threshold (Schmitt input)		0.7V_{DD}		V_{DD}+0.5	V
HYS	Hysteresis voltage			0.2 V_{DD}		V
V_{OL}	Output low voltage all ports[5, 9]	I_{OL} = 3.2 mA, V_{DD} = 2.7 V			0.4	V
V_{OL1}	Output low voltage all ports[5, 9]	I_{OL} = 20 mA, V_{DD} = 2.7 V			1.0	V
V_{OH}	Output high voltage, all ports[3]	I_{OH} = –20 µA, V_{DD} = 2.7 V	V_{DD}–0.7			V
		I_{OH} = –30 µA, V_{DD} = 4.5 V	V_{DD}–0.7			V
V_{OH1}	Output high voltage, all ports[4]	I_{OH} = –1.0 mA, V_{DD} = 2.7 V	V_{DD}–0.7			V
C_{IO}	Input/Output pin capacitance[10]				15	pF
I_{IL}	Logical 0 input current, all ports[8]	V_{IN} = 0.4 V			–50	µA
I_{LI}	Input leakage current, all ports[7]	V_{IN} = V_{IL} or V_{IH}			±2	µA
I_{TL}	Logical 1 to 0 transition current, all ports[3, 6]	V_{IN} = 1.5 V at V_{DD} = 3.0 V	–30		–250	µA
		V_{IN} = 2.0 V at V_{DD} = 5.5 V	–150		–650	µA
R_{RST}	Internal reset pull-up resistor		40		225	kΩ
V_{BOLOW}	Brownout trip voltage with BOV = 1[12]		2.35		2.69	V
V_{BOHI}	Brownout trip voltage with BOV = 0		3.45		3.99	V
V_{REF}	Reference voltage		1.11	1.26	1.41	V
t_C (V_{REF})	Temperature coefficient			tbd		ppm/°C
SS	Supply sensitivity			tbd		%/V

NOTES:

1. Typical ratings are not guaranteed. The values listed are at room temperature, 5 V.
2. See other Figures for details. Active mode: $I_{CC(MAX)}$ = tbd; Idle mode: $I_{CC(MAX)}$ = tbd
3. Ports in quasi-bidirectional mode with weak pull-up (applies to all port pins with pull-ups). Does not apply to open drain pins.
4. Ports in PUSH-PULL mode. Does not apply to open drain pins.
5. In all output modes except high impedance mode.
6. Port pins source a transition current when used in quasi-bidirectional mode and externally driven from 1 to 0. This current is highest when V_{IN} is approximately 2 V.
7. Measured with port in high impedance mode.
8. Measured with port in quasi-bidirectional mode.
9. Under steady state (non-transient) conditions, I_{OL} must be externally limited as follows:
   Maximum I_{OL} per port pin: 20 mA
   Maximum total I_{OL} for all outputs: 80 mA
   Maximum total I_{OH} for all outputs: 5 mA
   If I_{OL} exceeds the test condition, V_{OL} may exceed the related specification. Pins are not guaranteed to sink current greater than the listed test conditions.
10. Pin capacitance is characterized but not tested.
11. The I_{DD}, I_{ID}, and I_{PD} specifications are measured using an external clock with the following functions disabled: comparators, brownout detect, and watchdog timer. For V_{DD} = 3 V, LPEP = 1. Refer to the appropriate figures on the following pages for additional current drawn by each of these functions and detailed graphs for other frequency and voltage combinations.
12. Devices initially operating at V_{DD} = 2.7 V or above, and at f_{OSC} = 10 MHz or less, are guaranteed to continue to execute instructions correctly at the brownout trip point. Initial power-on operation below V_{DD} = 2.7 V is not guaranteed.
13. Devices initially operating at V_{DD} = 4.0 V or above and at f_{OSC} = 20 MHz or less are guaranteed to continue to execute instructions correctly at the brownout trip point. Initial power-on operation below V_{DD} = 4.0 V and f_{OSC} > 10 MHz is not guaranteed.

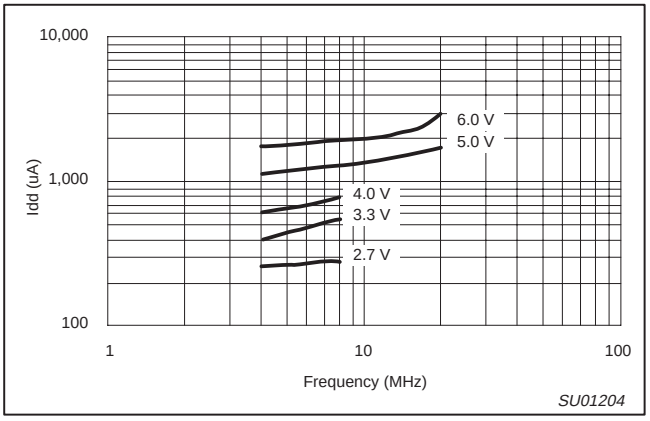


Figure 42. Typical Idd versus frequency (high frequency oscillator, 25°C)

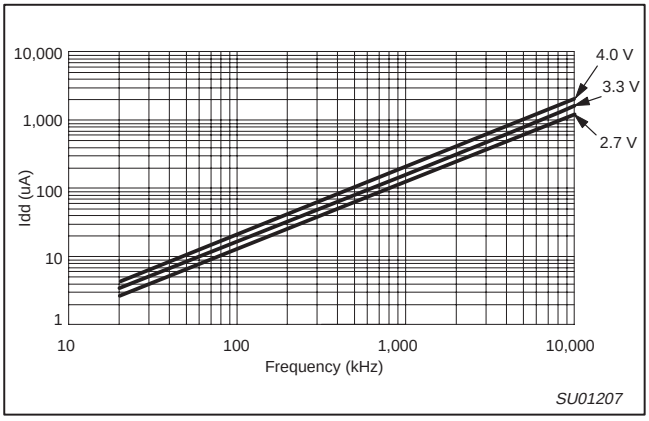


Figure 45. Typical Idle Idd versus frequency (external clock, 25°C, LPEP=1)

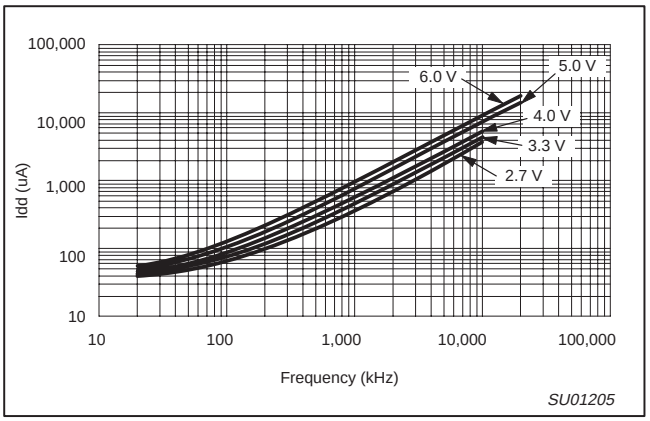


Figure 43. Typical Active Idd versus frequency (external clock, 25°C, LPEP=0)

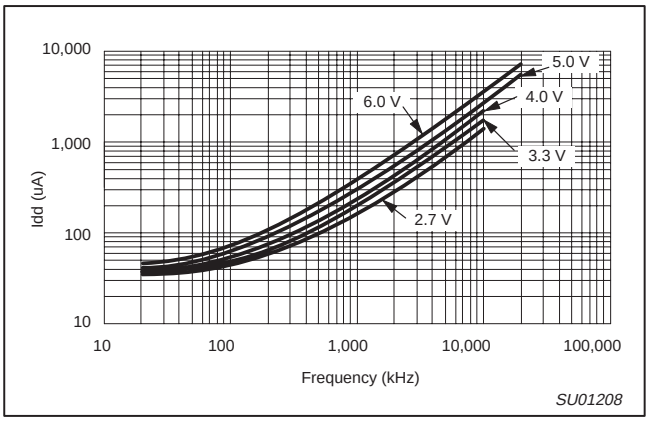


Figure 46. Typical Idle Idd versus frequency (external clock, 25°C, LPEP=0)

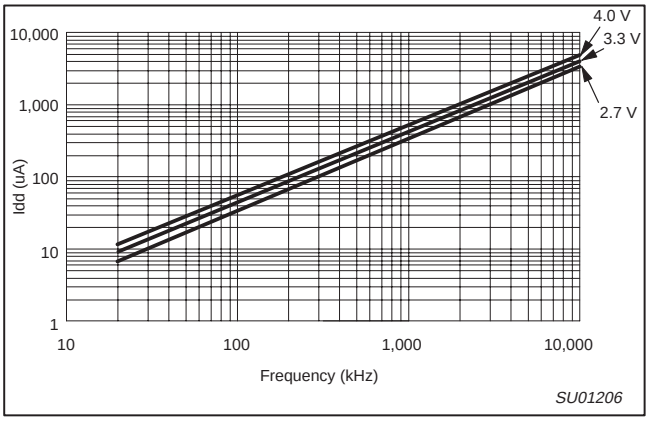
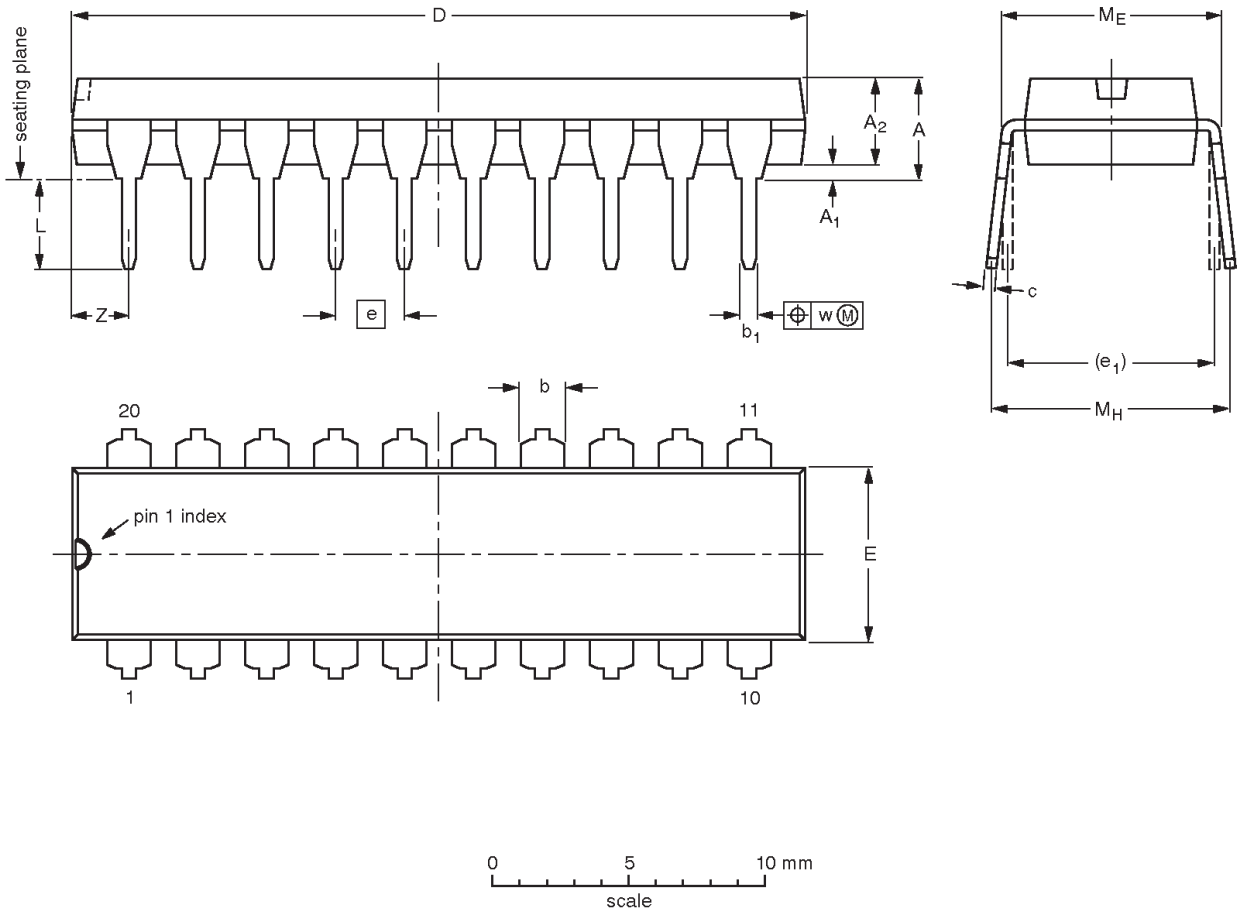


Figure 44. Typical Active Idd versus frequency (external clock, 25°C, LPEP=1)

DIP20: plastic dual in-line package; 20 leads (300 mil) **SOT146-1**



DIMENSIONS (inch dimensions are derived from the original mm dimensions)

UNIT	A max.	A_1 min.	A_2 max.	b	b_1	c	D (1)	E (1)	e	e_1	L	M_E	M_H	w	Z (1) max.
mm	4.2	0.51	3.2	1.73 1.30	0.53 0.38	0.36 0.23	26.92 26.54	6.40 6.22	2.54	7.62	3.60 3.05	8.25 7.80	10.0 8.3	0.254	2.0
inches	0.17	0.020	0.13	0.068 0.051	0.021 0.015	0.014 0.009	1.060 1.045	0.25 0.24	0.10	0.30	0.14 0.12	0.32 0.31	0.39 0.33	0.01	0.078

Note

1. Plastic or metal protrusions of 0.25 mm maximum per side are not included.

OUTLINE VERSION	REFERENCES				EUROPEAN PROJECTION	ISSUE DATE
	IEC	JEDEC	EIAJ			
SOT146-1		MS-001	SC-603			~~95-05-24~~ 99-12-27